Ad Hoc Networks 105 (2020) 102176

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Research on Crowdsourcing network indoor localization based on Co-Forest and Bayesian Compressed Sensing

Min Zhao, Danyang Qin*, Ruolin Guo, Guangchao Xu

Heilongjiang University, Harbin, 150080, PR China

ARTICLE INFO

Article history:
Received 25 November 2019
Revised 23 February 2020
Accepted 12 April 2020
Available online 12 May 2020

Keywords:
Indoor localization
Co-Forest
Bayesian compressed sensing
Crowdsourcing

ABSTRACT

Indoor Localization Technology (ILT) based on Wi-Fi network has been rapidly developed with high localization accuracy and low hardware requirements. Collecting the Received Signal Strength (RSS) samples to construct the fingerprint database, however, is time consuming and labor intensive, hindering the application of the technology. An Indoor Localization Method is proposed based on Co-Forest and Bayesian Compressed Sensing (ILM-CFBCS), utilizing the crowdsourcing network technology to collect RSS data and the min-max method to preprocess the data so as to establish the indoor fingerprint database. The user's position is determined according to the decision result of the random forest classifier trained by the Co-Forest algorithm combining with majority principle. Finally, a constructing method of offline fingerprint database is put forward by combining the similarity between Bayesian compressed sensing theory and reference point fingerprint to realize the fingerprint database update. The experimental results show that the proposed method can achieve good localization performance by using a small amount of data with labeled positions.

1. Introduction

With the development and optimization of mobile ad hoc network [1], indoor localization information in the fields of navigation services, mobile socialization, public safety, and smart city construction is extremely important for individuals [2], which uses radio communication network technologies and external localization methods to make the ILT gradually become a hot topic. Wi-Fi localization system, Bluetooth localization system, RF localization system, infrared localization system and ultrasonic localization system are commonly used [3]. Wi-Fi localization networks are rapidly spreading around the world with high speed and low cost. And Wi-Fi has become a very attractive wireless networks technology for localization because it is widely distributed in various buildings such as homes, shopping malls, and hospitals [4]. The fixed Access Points (APs) are a part of a Wi-Fi network system, which are typically distributed in indoors and the positions of the distribution are known. Mobile devices such as laptops and smartphones that can connect to Wi-Fi network to communicate directly or communicate through an AP to locate. Since Wi-Fi signals are not specifically used for location, single antennas, small bandwidths and complex indoor signal propagation environments make traditional localization methods such as distance-based ranging methods and signal angle-based inoperable. The Wi-Fi-based position fingerprint method [5] has gradually become the main research method of ILT. The position fingerprint identification method compares and matches the signal strength vector of each hot spot received by the mobile terminal in real time with the data stored in the fingerprint database after collecting signals, recording data and establishing databases, etc., thereby accurately locating the user's position.

However, a large number of fingerprint indoor localization technologies based on Wi-Fi network cannot meet the requirements of current indoor localization for accuracy and timeliness, due to the large amount of data collection, long time-consuming, the database update without considering and other issues. The method based on Co-Forest is proposed, focusing on the time-consuming and labor-intensive of existing fingerprint indoor localization and the requirements for accuracy, cost and real-time. The overall content architecture of this article is shown in Fig. 1.

The main contributions of this paper are as follows:

(1) A semi-supervised algorithm Co-Forest is used to achieve real-time location for user's positions by a small amount of data with labeled positions.
(2) The crowdsourcing technology is used to collect data and the min-max pre-processing method is used to solve the device heterogeneity problem.
(3) The step for updating the fingerprint database is added, combining the similarity between Bayesian compressed sensing theory and reference point fingerprint. The proposed system is

* Corresponding author.
E-mail address: qindanyang@hlju.edu.cn (D. Qin).

https://doi.org/10.1016/j.adhoc.2020.102176
1570-8705/

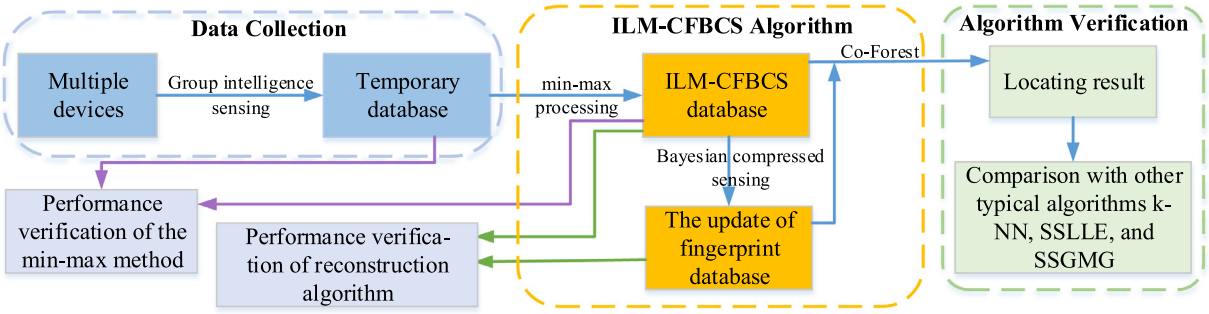


Fig. 1. The overall architecture of the paper.

Table 1
Comparison of advantages and disadvantages of different localization methods.

Method	Index	Advantage	Disadvantage
ToA	time	high precision	line of sight, strict time synchronization, the known AP position, special hardware equipment
TDoA	time difference	high precision, without needing strict time synchronization	line of sight, the known AP position, special hardware equipment
AoA	angle	without needing time synchronization, only need at least two receivers	line of sight, the known AP position, special hardware equipment
RSS	signal strength	low cost, without needing to know AP position	large environmental interference

easy to implement and without needing any additional infrastructure, providing the possibility for fast, scalable location.

The rest of the paper is organized as follows: Section 2 describes the existing indoor network localization methods based on Wi-Fi and machine learning. The Section 3 describes the offline processing phase, online localization phase and fingerprint database updating of ILM-CFBCS in detail. The Sction 4 designs simulation experiments to verify the good performance of the proposed method in terms of localization accuracy, device heterogeneity and database update. Section 5 summarizes our conclusions.

2. Related work

With the development of mobile communication network technology [6] and the increasing demand for position, the corresponding position service application is also developing [7]. Position Based Service (PBS) generally refers to providing services related to position information to current users through radio communication network technologies and external localization methods.

The relatively mature localization systems include ILT based on GPS-assisted, ILT based on magnetic field information matching, and ILT based on indoor wireless network communication. ILT based on GPS-assisted [8-10] has problems such as relatively low localization accuracy and high cost. ILT based on magnetic field information matching [11-13] has problems that are susceptible to interference from many external factors and are high cost. Indoor localization technologies based on indoor wireless network communication include infrared [14], ultrasonic [15], radio frequency identification [16], and Wi-Fi. Among them, Wi-Fi is widely applied for indoor localization because it is widely distributed and without requiring additional hardware investments.

With the development of Wi-Fi ILT, different localization methods have been proposed, as shown in Table 1. Among them, localization methods based on the Angle of Arrival (AoA) [17], the Time of Arrival (ToA) [18] and the Time Difference of Arrival (TDoA) [19] require special hardware devices, line-of-sight propagation, and need to know the position information of the APs in advance, causing the high cost, the difficult in implementation, and uneasy in widespread popularity. However, the localization method based on Received Signal Strength (RSS) [20] has received extensive attention [21], because it makes full use of widely covered Wi-Fi signals and mobile intelligent device resources, and does not need to install other hardware facilities. The Wi-Fi indoor localization method based on RSS uses the signal strength of the AP as a position fingerprint, which has a broad research prospect because it is easy to implement and easy to integrate with other localization systems [22].

With the introduction of various localization algorithms, localization systems suitable for different scenarios are also emerging. The nearest neighbor method (k-NN, WKNN), naive Bayesian method, neural network method and support vector regression method are common algorithms in position fingerprint localization system [23]. Ma et al. proposed an improved weighted fusion algorithm based on WKNN and probability method, but did not improve the time-consuming and labor-intensive problem of building a fingerprint database [24]. Brunato et al. used a Support Vector Machine (SVM) based on a Radial Basis Function (RBF) as a kernel function for position estimation, which has good adaptability. However, the localization algorithm based on SVM is difficult to apply in the real environment because it is difficult to choose the optimal kernel function [25]. Gan et al. proposed an indoor position algorithm with multi-sensor fingerprinting and deep learning, using statistical models and ray tracing methods to construct large sample data for training. The algorithm has good stability and accuracy, but the update problem of the fingerprint database is not considered [26]. Jain et al. proposed an indoor wireless network method based on Semi-Supervised Local Linear Embedding (SSLLE), using local linear embedding to reduce the dimensionality of the data and a semi-supervised learning algorithm to learn the radio map. But the performance of this method is affected by the parameter k, and there are some restrictions in the application [27]. Li et al. proposed a semi-supervised manifold-aligned fingerprint database construction algorithm based on global feature preservation (SSGMG). A small amount of labeled data with long acquisition time and a large number of easily collected unlabeled data are used to solve the positional calibration of unmarked data by solving the manifold alignment target function. Simultaneously, the

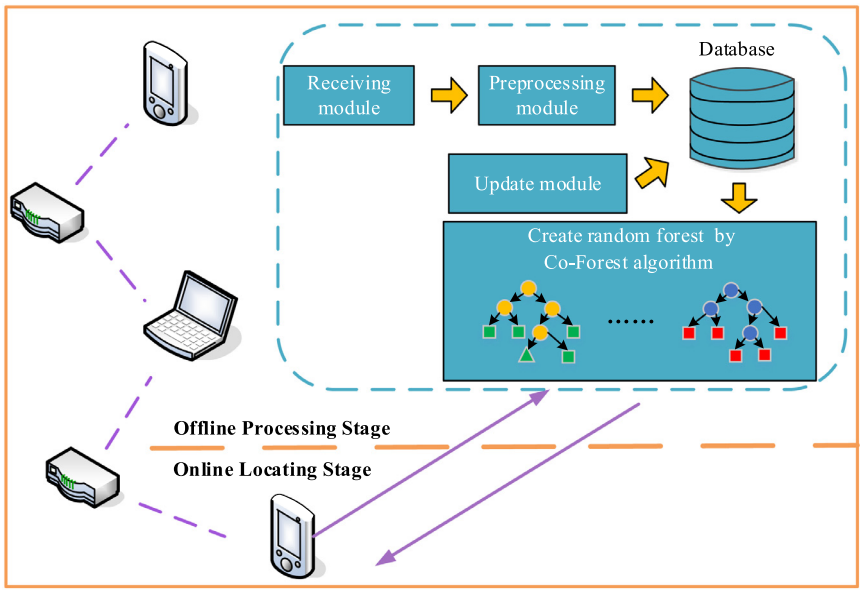


Fig. 2. The model of ILM-CFBCS.

global manifold structure is obtained by using the geodesic distance to fully exploit the corresponding features in a small amount of marker data, and the accuracy of the fingerprint database constructed under the condition of a small amount of labeled data is improved [28]. In addition, the compressed sensing theory for signal reconstruction is used for indoor location problems proposed by researchers at Beijing Jiaotong University for the first time, and the position of mobile terminals is reconstructed by compressed sensing [29]. This algorithm provides inspiration for the database reconstruction algorithm in this paper.

Based on the above research, an indoor localization method based on semi-supervised machine learning is proposed. The crowdsourcing technology is used to construct a fingerprint database with a small amount of labeled data. The decision results of the random forest classifier trained by the Co-Forest algorithm are used for location. Bayesian compressed sensing technology is used to update the database.

3. The proposed ILM-CFBCS

The proposed ILM-CFBCS system model is shown in Fig. 2, consisting of an offline processing phase and an online locating phase. The offline processing phase includes collecting data, building a database, and generating random forest classifiers trained by the Co-Forest algorithm. In the online locating phase, each tree in the random forest classifies the position of the querying user, the majority principle is used to determine the position information and the final decision result is fed to the querying user. In addition, the Bayesian compressed sensing technology is used to update the fingerprint database, to solve the problem of the difference between database data and real data due to environmental changes.

3.1. Offline processing stage of ILM-CFBCS

3.1.1. Data collection

The data collection process of ILM-CFBCS is divided into two parts, one is to carry out RSS data collection with labeled position, and the other is to collect RSS data without labeled position. In the collection with labeled position, the localization area is sampled and RSS samples are collected at each reference position for building a temporary database. The experimental spaces are divided and the reference positions are deployed. An RSS sample that records position coordinates and RSS values is collected by the staff at each reference position and stored as Eq. (1):

$$\boldsymbol{FP} = [fp_1, fp_1, \cdots, fp_N]^T \tag{1}$$

where $\boldsymbol{fp}_n = ([x_n, y_n], rss_n^1, rss_n^2, \ldots, rss_n^L)$ is the fingerprint at the reference position n ($n = 1, 2, \ldots, M$).

In the RSS data collection without labeled position, the volunteers that cover the entire localization areas are needed to select. The RSS samples without labeled position are collected from neighboring APs by sampling program installed in the mobile device of volunteers and are transmitted to the back-end fingerprint service. In the fingerprint service, the receiving module is always ready to receive RSS samples collected from volunteers and stores these samples in a temporary database. The storage forms of RSS data without labeled position are shown as to Eq. (2):

$$\boldsymbol{FP'} = \left[fp'_1, fp'_2, \cdots, fp'_1\right]^T \tag{2}$$

where $\boldsymbol{fp'}_i = [rss_i^1, rss_i^2, \cdots, rss_i^L]^T$ consists of the RSS value of the mobile terminal at the i-th reference position, and $[]^T$represents transpose.

3.1.2. Database construction

During the data collection process, the different mobile devices models used by the staff and volunteers will result in different RSS values, which will affect the localization accuracy. Therefore, it is necessary to normalize and discretize the RSS samples in the temporary database to build a database of ILM-CFBCS.

The min-max normalization method is chosen to normalize the sample data, which performs positive and negative normalization on all RSS samples in the temporary database, and limits the range to [-1, 1]. The RSS value of the AP in each sample can be predicted because of the linear transmission. Assume that the sampled set of RSS samples is defined as $Q=\{q_1,q_2,\ldots,q_n\}$, where the number of samples is n, each sample is defined as $q=\{o_1,o_2, o_i,\ldots,o_m\}$. The number of APs is m, and o_i represents the RSS value collected from the i-th AP. The sample is processed as shown in Eq. (3):

$$q' = round\left(\frac{q - ave(q)}{max(q) - min(q)}\right) \tag{3}$$

Table 2
Partial RSS samples (dBm) in the temporary database at a specific reference position.

Device	AP1	AP2	AP3	AP4	AP5	AP6
Device 1	-85	-42	-89	-77	-76	-89
Device 2	-81	-35	-76	-70	-77	-85
Device 3	-83	-40	-85	-84	-72	-90
Device 4	-67	-18	-64	-61	-53	-67
Device 5	-79	-35	-87	-72	-73	-81

Table 3
Signed results of RSS samples in the temporary database at a specific reference position.

Device	AP1	AP2	AP3	AP4	AP5	AP6
Device 1	-0.1	0.7	-0.2	0.1	0.1	-0.2
Device 2	-0.2	0.8	-0.1	0.1	-0.1	-0.2
Device 3	-0.1	0.8	-0.1	-0.1	0.2	-0.2
Device 4	-0.2	0.8	-0.1	-0.1	0.1	-0.2
Device 5	-0.1	0.7	-0.2	0.0	0.0	-0.1

where *max(q)* represents the maximum value of the RSS samples in *q; min(q)* represents the minimum value of the RSS samples in *q; min(q)* represents the average of the RSS values in *q*. The round function only keeps the RSS data value by one decimal place.

If decision trees in a random forest are created, the sample set by repeatedly splitting the attribute values are divided. The attributes are equivalent to Aps in this paper. The sample is divided into two subsets each time, one subset contains the samples whose attribute value is less than the split value, and the other subset contains the samples whose attribute value is greater than the split value. Because of the discretization, the number of attribute values is greatly reduced, avoiding repeated segmentation of continuous attributes, reducing computational overhead and improving the establishment efficiency of decision tree.

The RSS samples before and after processing are compared in Table 2 and Table 3. It can be seen that the normalization process reduces the absolute value difference of the RSS samples and greatly reduces the impact of different devices on the database establishment.

3.1.3. Generation of random forest classifiers

The process of collecting RSS data with labeled position is time consuming and labor intensive compared to use the volunteers for intelligent RSS data collection without labeled position. So, we want to locate the target by making full use of a large number of samples without labeled position, and without collecting the samples with labeled position. The RSS vectors is considered as a sample and the position information is considered as a classification label. A semi-supervised classification algorithm Co-Forest is introduced, which applies a large number of samples without labeled position to improve the tree classifier which is based on a small number of samples with labeled position.

Let L be the RSS sample set with labeled position and U be the sample set without labeled position. The random forest is denoted as $R=\{rl_i, i=1, ..., N\}$, and the set contains N decision trees rl_i. A new integrated classifier$R_i=\{R-rl_i, i=1, ..., N\}$is obtained by removing rl_i from R. R_i contains all decision trees except rl_i. R_i is the companion classifier for rl_i. Each rl_i has a corresponding companion classifier R_i. The key to Co-Forest is to process the sample set U without labeled position using the companion classifier R_i and select the samples and labels without labeled position with the highest confidence predicted by R_i, in order to supplement the set L with labeled position. The classifier rl_i is enhanced using an additional sample set with labeled position. The pseudo code of Co-Forest is shown in Table 4.

A random forest R consisting of N random trees is first constructed in the Co-Forest algorithm (Line 1). N random sample sets are obtained from L by repeatable sampling, and a decision tree is constructed from each sample set by using the improved CART algorithm. The CART pruning operation is cancelled to improve the randomness of each tree. These decision trees are continually modified by iteratively using samples without labeled position. Two steps are completed in each iteration: (1) a supplementary sample set L'_i is generated from U for each sub-classifier rl_i (Line 7-14); (2) each rl_i is corrected by applying the union of L and L'_i (Line 15-17). Each rl_i can obtain a random sample set U'_i from the sampled signal of U in step 1. Then, with the predicted label, the sample without labeled position with a higher confidence than the threshold is inserted into L'_i.

The ratio of the component classifiers that match the confidence of a sample without labeled position to the predicted result is defined in R_i. Then each component classifier in R_i can provide prediction results quickly, so the corresponding ratio can be obtained efficiently, which is a significant improvement in the collaborative training algorithm. Each rl_i of the random forest updates the generated decision tree by running CART on the union of L and L'_i. All trees are updated to run in parallel to improve speed. The above iteration runs until rl_i does not change. Finally, the corrected classifiers by random forest can be obtained. In addition, $e_{i,t}/e_{i,t-1} < 1$ must be met in order to ensure absolute convergence of the results of corrected classifiers. $e_{i,t}$ represents the classification error rate of the supplementary sample $L'_{i,t}$ of R_i at the i-th repetition, and the initial value of $e_{i,t}$is 0.5.

The classification error rate should decrease as the number of iterations increases. Therefore, the Line 10 and Line 16 are added as a judgment in the algorithm, and the subsequent operations are performed when the condition is satisfied.

In addition, N (the number of trees in a random forest) is an extremely important parameter for the Co-Forest algorithm. When N is very small, the performance of Co-Forest algorithm will improve with the increase of N. When N reaches a certain value, such performance improvement will not continue. The Out of Bag (OOB) estimation error is proved to be an unbiased error estimate for random forests [30]. The OOB estimation error (ε_{OOB}) is adopted as an indicator to measure the performance of random forests. The optimal value of N is determined by the variation of the ε_{OOB} with the increase of N value.

The OOB data and the OOB decision tree are first defined. Then assume that L is a sample set with labeled position. $R(i=1, 2, ..., n)$ is a repeatable sample set that is obtained by repeatable sampling from L in order to generate a random forest decision tree. The probability that each sample (x_j,y_j) (where $x_j,y_j \in L(j=1, 2, ..., n)$) in L is not selected is p ($p=(1-1/n)^n$, n is the number of samples in L). $p \approx 0.368$ when n is large enough, that is, about 1/3 of the samples in L are not in R, these samples are called OOB data and can be used to validate the model. Finally, a set of OOB decision trees called $x_j$are generated using these R. ε_{OOB} is defined as Eq. (4):

$$\varepsilon OOB = 1 - \frac{1}{n}\sum_{j=1}^{n} I_{(OOBtrees\_pr(x_j)=y_j)} \tag{4}$$

where $OOBtrees\_pr(x_j)$ represents the prediction result of OOB decision tree of x_j, which is determined by each tree by majority vote; $I_{(\,\cdot\,)}$ represents the function with value being 1 for the condition of the predicted result equal toy_j and 0 otherwise. ε_{OOB} is the average prediction error for each training sample x_j, and only the trees generated by R not containing x_j are utilized.

N is continuously adjusted and ε_{OOB} of the generated random forest is calculated in the original data set with labeled sample and the processed data set with labeled sample, in order to obtain the

Table 4
The pseudo code of Co-Forest.

ALGORITHM 1: Process of Co-Forest

Input: L(RSS_label $\neq \emptyset$),U(RSS_label $= \emptyset$),N(number_of_random_trees),θ(confidence_threshold), $\mathbb{C} \leftarrow confidence$, $\mathbb{S} \leftarrow boostrapSample$
1. Initial_Forest ← **select**$N \subset L$; // Create an initial random forest // of L tree classifiers from L
2. $t = 0$;
3. **for** $i = 1$: N **do**
4. $\{e_{i,t} = 0.5;\}$
5. **while** ($random\_trees \neq 0$) **do** // When the random tree changes
6. $\{t = t + 1;$
7. **for** $i = 1$: N **do**
8. $\{e_{i,t} = calculate\_Error\_Rate(R_i, L);$
9. $L'_{i,t} = NULL;$
10. **if**$(e_{i,t} < e_{i,t-1})$
11. $U'_{i,t} = \mathbb{S}(U);$
12. **for** $x \in U'_{i,t}$ **do**
13. $\{$**if**$(\mathbb{C}(R_{i,x} > \theta))$
14. $L'_{i,t} = L'_{i,t} \cup \{(x, R_i(x))\};\}\}$
15. **for** $i = 1$: N**do**
16. $\{$**if**$(e_{i,t} < e_{i,t-1})$
17. $rl_i = creat\_Random\_Tree(L'_{i,t} \cup L);\}\}$
18. **return** R
Output:$R(refined\_random\_trees)$ // Modified random tree

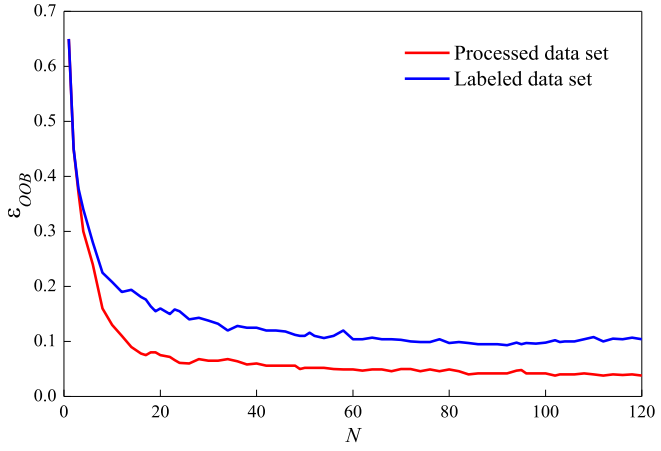


Fig. 3. The OOB estimation error of the processed labeled data set and the original labeled data set.

optimal value of N. ε_{OOB} varies with the number of trees in the random forest as shown in Fig. 3.

As can be seen from Fig. 3, ε_{OOB} will decrease sharply as the N increases when the N is small (less than 15). ε_{OOB} does not change significantly as the N increases when the N is greater than 30. Therefore, we use 30 as the optimal value of N in the experiment. In addition, ε_{OOB} of the processed data set is on average 5.9% lower than the ε_{OOB} of the original data set, because the normalization process reduces the impact of RSS sample changes on prediction accuracy.

3.2. Online locating phase of ILM-CFBCS

The online locating phase of ILM-CFBCS is relatively simple and fast, designing to fulfill the user's request for real-time location. The process of online locating is shown in Fig. 4. When the user gets the RSS value of a new position, the value is sent to the ILM-CFBCS system to ask the user's current position. Each tree in the random forest will compare and match the RSS value with the fingerprint database of the ILM-CFBCS, and the position with the highest probability is chosen to use for classifying the position of the querying user. The principle of the majority is used to determine the user's position. Finally, the final decision result is fed back to the requesting user.

3.3. The fingerprint database update of ILM-CFBCS

The distribution of the wireless signal in the position area changes when the transmit power or quantity of the AP changes, which will cause the difference between the fingerprint database with the actual environment. The fingerprint database needs to be updated or reconstructed to avoid the impact of this situation on the localization results. In the position area, the RSS from an AP is smoothly changed, and the Fourier coefficients corresponding to the fingerprint database are sparse [31]. Therefore, the compressed sensing theory is used to update the fingerprint database by collecting less fingerprint data with labeled position to reduce labor costs. The definitions of the main symbols involved in the update technique is shown in Table 5.

3.3.1. The construction of perceptual matrix

It is assumed that the fingerprint $\boldsymbol{fp}_n^i = ([x_n, y_n], rss_n^i)$ at the nth (n=1,2..., L_N) reference position is related to the RSS of other reference positions in the entire localization area, which is represented by the relationship matrix $\boldsymbol{\varphi}_n = [\varphi_n^1, \varphi_n^2, ..., \varphi_n^{L_N}]$, $\sum_{i=1}^{L_N} \varphi_n^i = 1, 0 \le \varphi_n^i \le 1$, representing the similarity of the n-th reference position to all other reference positions in the localization area. At each reference position, the position coordinates of the reference positions and the rss_n^i from the ith AP are obtained based on the fingerprint $fp_n^i = ([x_n, y_n], rss_n^i)$. The physical distance between the reference positions at adjacent positions closer is, the smaller the difference in received signal strength from the same AP will be. The position coordinates [x_n,y_n] and the received signal strength rss_n^i are as the influencing factors of the reference position similarity, to better measure the similarity between the different reference positions in the localization area. The two different factors need to be normalized separately, because they need to be combined to measure the similarity.

Let d_n^i be the physical distance between the reference position n (n=1,2,..., C_N) and the reference position i (i=1,2,..., L_N), and calculate the d_n^i by Euclidean distance as shown in Eq. (5):

$$d_n^i = \sqrt{(x_{nc} - x_n)^2 + (y_{nc} - y_n)^2} \tag{5}$$

The distance vector between the n-th reference position and all other reference positions in the localization area is $\boldsymbol{d}_n =$

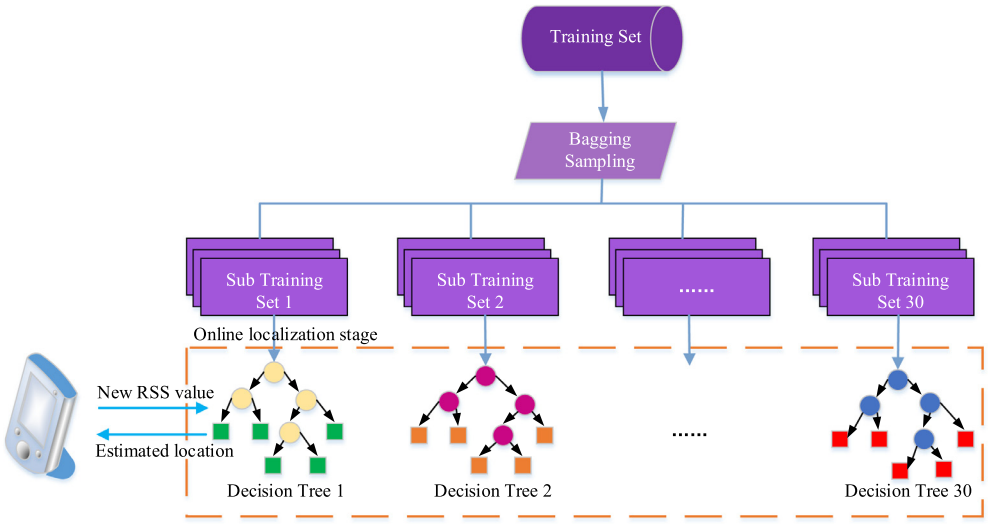


Fig. 4. The process of online location.

Table 5
Explanations for the main parameters in this section.

Symbol	Explanations
$\boldsymbol{fp}_o$	Fingerprint in the original fingerprint database
$\boldsymbol{fp}_c$	Fingerprint after environmental change
$\Delta\boldsymbol{fp}$	Vector of the actual change of the fingerprint
$\Delta\widetilde{\boldsymbol{fp}}$	Fingerprint change value vector at *CN*positions
LN	Number of reference positions in the position area
CN	Number of newly collected fingerprints after environmental changes
$\boldsymbol{X}$	Measurement noise matrix
$\boldsymbol{W}$	Perceptual matrix
$\boldsymbol{\mu}$	Mean vector of $\Delta\boldsymbol{fp}$
$\boldsymbol{C}_{\text{cov}}$	Covariance matrix of $\Delta\boldsymbol{fp}$
$\boldsymbol{\theta}$, λ	Hyperparameters

$[d_n^1, d_n^2, \ldots, d_n^{L_N}]$. d_n^i is normalized using all distances as shown in Eq. (6):

$$Dis_n^i = \frac{d_n^i}{\sum_{i=1}^{N} d_n^i} \tag{6}$$

where Dis_n^i represents the distance similarity after the normalization process. The closer the distance between two reference positions is, the more similar the environments as well as the greater the correlations will be.

The RSS received from the AP at a certain position is unstable and fluctuating due to the influence of the indoor environment, using the RSS fingerprint mean to calculate the similarity between the measured position and the reference position introduces spatially non-adjacent reference positions, which leads to localization errors. To this end, He et al. proposed to include the RSS variance from an AP at a certain position into the rules for similarity calculation. And verified that the similarity measure of the introduced variance is better than the Euclidean distance and the cosine similarity [32]. The mean and variance of RSS at all reference positions in the experimental environment are calculated, and the reference position sampled after the environmental change are assumed as the measured position. The variance for calculating the similarity between the measured position and all other reference positions is introduced.

Let Sim_n^i be the similarity measure of the RSS between the measured position n (n=1,2,..., C_N) and the reference position i (i=1,2,..., L_N). The variance $(\delta_o^i)^2$ of the RSS is introduced. The similarity between the two RSS fingerprints is calculated as shown in Eq. (7):

$$Sim_n^i = \left(rss_o^n - \overline{rss}_o^i\right)^2 + \left(\delta_o^i\right)^2 \tag{7}$$

where $\overline{rss}_o^i$ is the mean of the RSS at the i-th reference position. The difference of RSS is also normalized, as shown in Eq. (8):

$$Sim_n^i = \frac{Sim_n^i}{\sum_{i=1}^{L_N} Sim_n^i} \tag{8}$$

where Sim_n^i represents the similarity of the RSS after the normalization process. The smaller Sim_n^i is, the closer the RSS between the two positions is, and the greater the possibility of experiencing the same RSS changing will be when the environment changes. The Dis_n^i and Sim_n^i are combined as the similarity measure equation to measure the similarity between the measured position n (n=1,2,..., C_N) and the reference position i (i=1,2,..., L_N), as shown in Eq. (9):

$$C_n^i = e^{-\left(aDis_n^i\right)^2 - (1-a)^2\left(Sim_n^i\right)^2} \tag{9}$$

where a represents the weight in the similarity calculation of physical distance Dis_n^i and RSS difference Sim_n^i. The similarity φ_n^i between the measured position n (n=1,2,..., C_N) and the reference position i (i=1,2,..., L_N) is obtained by performing the normalization for, as shown in Eq. (10):

$$\varphi_n^i = \frac{C_n^i}{\sum_{i=1}^{L_N} C_n^i} \tag{10}$$

The greater the φ_n^i is, the higher the correlation between the changing value $\Delta\widetilde{rss}^n$ and the actual changing value Δrss^n will be.

Similarly, the similarity with other reference positions is calculated by Eq. (9) for all reference positions endowed a new fingerprint. The perception matrix based on similarity is obtained as shown in Eq. (11):

$$\boldsymbol{W} = [w_1, w_2, ..., w_{C_N}]^T \tag{11}$$

3.3.2. Bayesian compressed sensing reconstruction algorithm

Assume that the vector of received signal strengths of L_N reference positions covered by an AP in the localization area is $\boldsymbol{fp}_o = (rss_o^1, rss_o^2, ..., rss_o^{L_N})^T$. The RSS vector $\boldsymbol{fp}_c = (rss_c^1, rss_c^2, ..., rss_c^{L_N})^T$ becomes when the environment of localization area changes. The vector of RSS actual change value obtained by the calculation is $\Delta\boldsymbol{fp} = (\Delta rss^1, \Delta rss^2, ..., \Delta rss^{L_N})^T = \boldsymbol{fp}_c - \boldsymbol{fp}_o$. Assume that the RSS vector after the environmental change collected at the C_N ($C_N < L_N$) reference positions is $\boldsymbol{fp}_c = (rss_c^1, rss_c^2, ..., rss_c^{C_N})^T$, then the vector of RSS fingerprint change value of the C_N reference positions is $\Delta\widetilde{\boldsymbol{fp}} = (\Delta\widetilde{rss}^1, \Delta\widetilde{rss}^2, ..., \Delta\widetilde{rss}^{C_N})^T$. As shown in Eq. (12):

$$\Delta\widetilde{\boldsymbol{fp}} = \boldsymbol{W}\Delta\boldsymbol{fp} + \boldsymbol{X} \tag{12}$$

where $\boldsymbol{X} = [x_1, x_2, ..., x_{C_N}]^T$ is the Gaussian ambient noise with a mean of 0 and a variance of δ^2. According to Bayesian Compressed Sensing Theory, $\Delta\boldsymbol{fp}$ is reconstructed using $\Delta\boldsymbol{fp}$ and the perceptual matrix $\boldsymbol{W}$ constructed as described in section 3.3.1, as shown in Eq. (13):

$$\begin{bmatrix} \Delta\widetilde{rss}^1 \\ \Delta\widetilde{rss}^2 \\ \vdots \\ \Delta\widetilde{rss}^{C_N} \end{bmatrix} = \begin{bmatrix} w_1^1 & w_1^2 & \cdots & w_1^{L_N} \\ w_2^1 & w_2^2 & \cdots & w_2^{L_N} \\ \vdots & \vdots & & \vdots \\ w_{C_N}^1 & w_{C_N}^2 & \cdots & w_{C_N}^{L_N} \end{bmatrix} \begin{bmatrix} \Delta rss^1 \\ \Delta rss^2 \\ \vdots \\ \Delta rss^{L_N} \end{bmatrix} + \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{C_N} \end{bmatrix} \tag{13}$$

Assume that the RSS value at each reference position approximates a Gaussian distribution, as shown in Eq. (14):

$$rss_o^n \sim N\left(\overline{rss_o^n}, \delta_o^{n2}\right), n = 1, 2, ..., L_N \tag{14}$$

where $\overline{rss_o^n}$ represents the mean of the RSS at reference position n and δ_o^{n2} represents the variance. The Gaussian distribution is additive, so $\Delta\widetilde{rss}^n = rss_c^n - rss_o^n$ still obeys the Gaussian distribution, as shown in Eq. (15):

$$\Delta rss^n \sim N\left(\Delta\overline{rss^n}, \delta_o^{n2} + \delta_c^{n2}\right), n = 1, 2, ..., L_N \tag{15}$$

where $\Delta\overline{rss^n}$ represents the mean of Δrss^n.

The Gaussian likelihood function of the RSS change vector $\Delta\boldsymbol{fp}$ and the noise variance δ^2 can be obtained according to Eq. (15), as shown in Eq. (16):

$$p\left(\Delta\widetilde{\boldsymbol{fp}}|\Delta\boldsymbol{fp}, \delta^2\right) = \frac{1}{\left(2\pi\delta^2\right)^{\frac{C_N}{2}}} e\left(-\frac{\| \Delta\widetilde{\boldsymbol{fp}} - \boldsymbol{W}\Delta\boldsymbol{fp}\|^2}{2\delta^2}\right) \tag{16}$$

According to the Bayesian probability theory, the posterior probability distribution $p(\Delta\boldsymbol{fp}|\Delta\widetilde{\boldsymbol{fp}}, \delta^2)$ of $\Delta\boldsymbol{fp}$ can be found by the known prior probability distribution $p(\Delta\boldsymbol{fp})$ and the observation vector $\Delta\widetilde{\boldsymbol{fp}}$ of $\Delta\boldsymbol{fp}$. The reconstruction problem of $\Delta\boldsymbol{fp}$is transformed into the distribution problem of $\Delta\boldsymbol{fp}$when the posterior probability of $\Delta\boldsymbol{fp}$is the largest. As shown in Eq. (17):

$$\begin{aligned} &\arg\max_{\Delta\boldsymbol{fp}} p\left(\Delta\boldsymbol{fp}|\Delta\widetilde{\boldsymbol{fp}}, \delta^2\right) \\ &s.t. \quad \Delta\boldsymbol{fp} \sim p(\Delta\boldsymbol{fp}) \end{aligned} \tag{17}$$

According to (14) and (16), the posterior probability distribution of $\Delta\boldsymbol{fp}$ still satisfies the Gaussian distribution shown in Eq. (18):

$$p\left(\Delta\boldsymbol{fp}|\Delta\widetilde{\boldsymbol{fp}}, \delta^2\right) \sim N(\boldsymbol{\mu}, \boldsymbol{C}_{\text{cov}}) \tag{18}$$

where $\boldsymbol{\mu}$ is the mean vector of the actual changing vector $\Delta\boldsymbol{fp}$ of the signal strength, and $\boldsymbol{C}_{\text{cov}}$ is the covariance matrix of $\Delta\boldsymbol{fp}$. Once the mean and variance of the posterior probability distribution are found, the posterior probability distribution can be determined to achieve the reconstruction of $\Delta\boldsymbol{fp}$. The RSS of the localization area is reconstructed according to Eq. (19), and the update of the offline fingerprint database is implemented.

$$\boldsymbol{fp}_n = \boldsymbol{fp}_o + \Delta\widetilde{\boldsymbol{fp}} \tag{19}$$

where $\boldsymbol{fp}_c$ represents the reconstructed RSS vector.

Let $\theta_n = (\delta_o^{n2} + \delta_c^{n2})^{-1}$ in Eq. (17), and $\boldsymbol{\theta} = [\theta_1, \theta_2, ..., \theta_{L_N}]$, then Eq. (15) can be further expressed as Eq. (20):

$$\Delta rss^n \sim N\left(\Delta\overline{rss^n}, \theta_n^{-1}\right), n = 1, 2, ..., L_N \tag{20}$$

A Gaussian prior distribution of $\Delta\boldsymbol{fp}$ can be gotten, as shown in Eq. (21):

$$p(\Delta\boldsymbol{fp}|\boldsymbol{\theta}) \sim \prod_i^N N(\Delta\boldsymbol{fp}|\theta_i^{-1}) \tag{21}$$

The posterior distribution $p(\Delta\boldsymbol{fp}, \boldsymbol{\theta}, \lambda|\Delta\widetilde{\boldsymbol{fp}})$ of $\Delta\boldsymbol{fp}$, $\boldsymbol{\theta}$ and λ under the condition of known $\Delta\widetilde{\boldsymbol{fp}}$ can be obtained as shown in Eq. (22):

$$\begin{aligned} p(\Delta\boldsymbol{fp}, \boldsymbol{\theta}, \delta^2|\Delta\widetilde{\boldsymbol{fp}}) &= \frac{p(\Delta\widetilde{\boldsymbol{fp}}|\boldsymbol{\theta}, \lambda, \Delta\boldsymbol{fp})p(\Delta\boldsymbol{fp}, \boldsymbol{\theta}, \delta^2)}{p(\Delta\widetilde{\boldsymbol{fp}})} \\ &= p(\Delta\boldsymbol{fp}|\boldsymbol{\theta}, \delta^2, \Delta\widetilde{\boldsymbol{fp}})p(\boldsymbol{\theta}, \delta^2|\Delta\widetilde{\boldsymbol{fp}}) \end{aligned} \tag{22}$$

The posterior probability distribution $p(\Delta\boldsymbol{fp}|\Delta\widetilde{\boldsymbol{fp}}, \delta^2)$ of $\Delta\boldsymbol{fp}$ is obtained as shown in Eq. (23):

$$\begin{aligned} p(\Delta\boldsymbol{fp}|\boldsymbol{\theta}, \delta^2, \Delta\widetilde{\boldsymbol{fp}}) &= \frac{p(\Delta\widetilde{\boldsymbol{fp}}|\Delta\boldsymbol{fp}, \delta^2)p(\Delta\boldsymbol{fp}|\boldsymbol{\theta})}{p(\Delta\widetilde{\boldsymbol{fp}}|\boldsymbol{\theta}, \delta^2)} \\ &= (2\pi)^{-\frac{L_N}{2}} |\boldsymbol{C}_{\text{cov}}|^{-\frac{1}{2}} e^{\left(-\frac{1}{2}(\Delta\boldsymbol{fp}-\boldsymbol{\mu})^T \boldsymbol{C}_{\text{cov}}^{-1}(\Delta\boldsymbol{fp}-\boldsymbol{\mu})\right)} \end{aligned} \tag{23}$$

where the posterior mean $\boldsymbol{\mu}$ and the covariance $\boldsymbol{C}_{\text{cov}}$ are expressed as Eq. (24):

$$\begin{aligned} \boldsymbol{\mu} &= \delta^{-2}\boldsymbol{C}_{\text{cov}}\boldsymbol{W}^T\Delta\widetilde{\boldsymbol{fp}} \\ \boldsymbol{C}_{\text{cov}} &= (\delta^{-2}\boldsymbol{W}^T\boldsymbol{W} + \boldsymbol{\Lambda})^{-1} \end{aligned} \tag{24}$$

where $\boldsymbol{\Lambda} = diag(\theta_1, \theta_2, ..., \theta_N)$. Let $\lambda = \delta^2$, the mean $\boldsymbol{\mu}$ and the covariance $\boldsymbol{C}_{\text{cov}}$ are functions of $\boldsymbol{\theta}$, so the problem is transformed into finding $\boldsymbol{\theta}$, λ when $p(\boldsymbol{\theta}, \lambda|\Delta\widetilde{\boldsymbol{fp}})$ is maximized. In order to find the $\boldsymbol{\theta}$, λ, the second kind of maximum likelihood estimation function is used [33]. Let C_{nn}be the n-th value on the diagonal of $\boldsymbol{C}_{\text{cov}}$, and the updated value θ'_n of θ_n is calculated as shown in Eq. (25):

$$\theta'_n = \frac{1 - \theta_n C_{nn}}{\mu_n^2} \tag{25}$$

where μ_n is the n-th element of vector $\boldsymbol{\mu}$. The same is available:

$$\lambda' = \frac{\left\| \Delta\widetilde{\boldsymbol{fp}} - \boldsymbol{W}\boldsymbol{\mu} \right\|^2}{C_N - W_n(1 - \theta_n W_{nn})} \tag{26}$$

The mean $\boldsymbol{\mu}$ and the covariance $\boldsymbol{C}_{\text{cov}}$ can be calculated by the Eq. (24) after assigning the initial values to $\boldsymbol{\theta}$, λ. Based on the Eq. (25) and (26), the obtained values of the mean $\boldsymbol{\mu}$ and the covariance $\boldsymbol{C}_{\text{cov}}$by calculating can be used to calculate the updated value of $\boldsymbol{\theta}$ and λ, so loop iterations until convergence.

In summary, the Bayesian Compressed Sensing Theory is used to update the offline fingerprint database by reconstructing the RSS change vector under the premise of known $\Delta\widetilde{\boldsymbol{fp}}$ and initial fingerprint database.

4. Experiment and analysis

The indoor layout of experimental site is shown in Fig. 5, which is the 7th floor of the Physics and Electronics Laboratory Building (PELB) of Heilongjiang University, including typical indoor feature

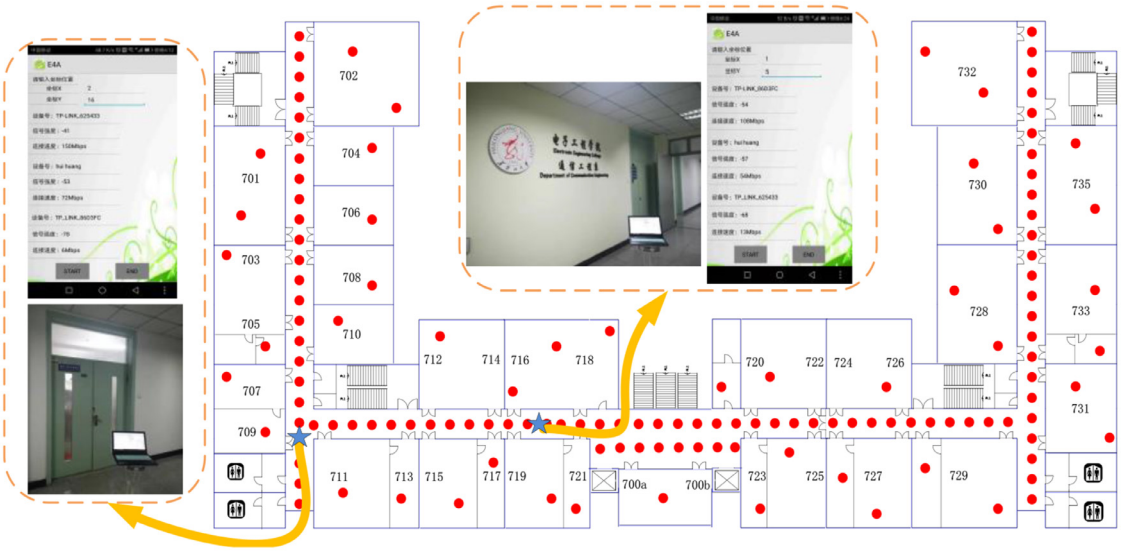


Fig. 5. Floor plan of the 7th floor of the PELB.

environments such as student laboratories, classrooms, teacher offices, and corridors. The red circle indicates the reference position in Fig. 5, a reference position is deployed along the centerline every 3 m in the corridor; the area to be located is divided into 2.5m × 2.5m grids and the reference positions are deployed in the four corners of the grid in the hallway hall; a different number of reference positions are deployed depending on the dimension of the room in each room.

Two samples with labeled position and without labeled position were collected using the smartphone sampling program. Four staff members carried different mobile devices to collect 20 RSS samples with labeled position at 2Hz at each reference position. Samples are sampled by a self-developed sampling program installed on the mobile device, and position coordinates are input through the user interface. Four different devices used include a Dell laptop, a Samsung smartphone, a Huawei smartphone and a Lenovo tablet. The sampling program includes two versions running on Android and Windows systems. Eighty samples with labeled position were collected at each reference position and a total of 19,200 samples with labeled position were collected throughout the experimental site. Only a small number of samples with labeled position and some samples without labeled position were used by the proposed method.

In order to obtain the samples without labeled position, a volunteer is selected from each laboratory throughout the experimental site so that the sample without labeled position could be dispersed throughout the experimental site. When volunteers are working in the lab, the sampling program in volunteers' mobile device collects RSS samples without labeled position from surrounding APs in the background and uploads to the fingerprint service. This implicit collection process does not require active interaction by the user, and the server sends information to control the start and the end of the collection.

4.1. The analysis of data collection differences

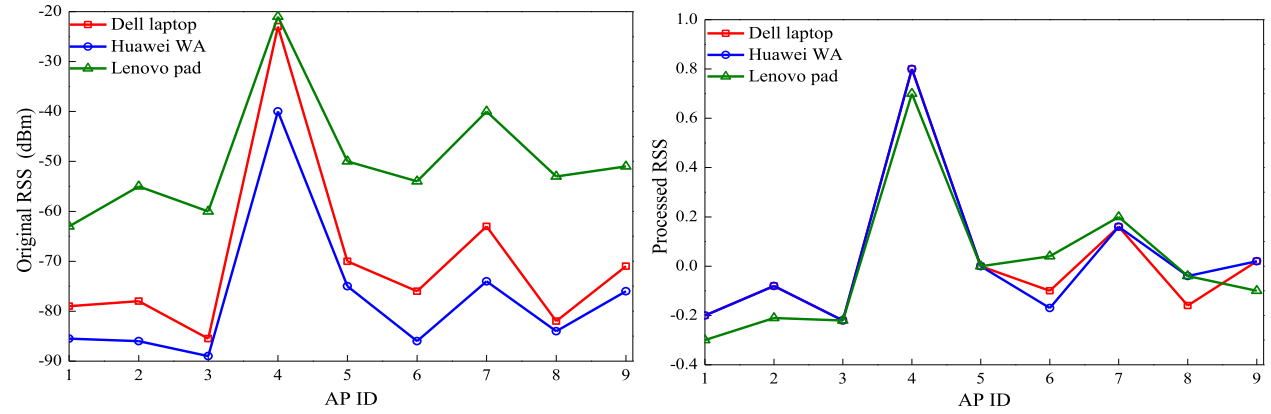
4.1.1. Processing of data differences

A set of samples provided by different devices is collected at each fixed reference position for analyzing the characteristics of RSS samples from different types of devices. The mobile devices with different software and hardware are selected to make the experimental data persuasive: A Dell I5 set-up laptop running Windows 10, a Huawei WAS-AL00 (Huawei WA) powered by Android 8.0, and a Lenovo small tablet supported by Android 7.0. A Lenovo Air13 laptop running Windows 7 system, a Letv X500 powered by Android 6.0, and a Huawei nova 2s (Huawei n2s) powered by Android 9.0.

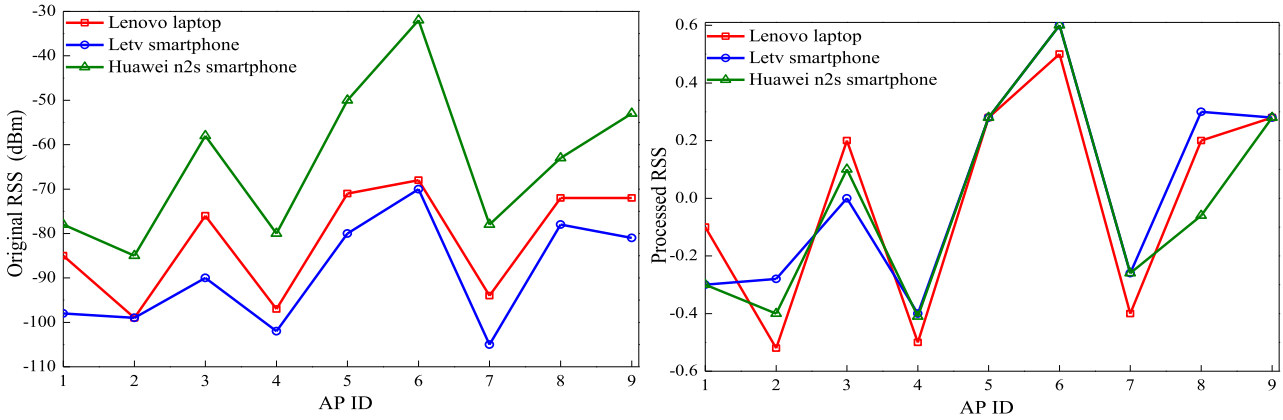
The RSS samples are collected from the APs around several reference positions selected in the PELB by the above mobile device, then the sampled data are normalized and discretized. The changes between the original sample and the processed sample are shown in Fig. 6. The changes in the original RSS values of the AP of around reference positions 2 and 4 are shown in Figure a), c), respectively. Fig. 6 b), d) are the results of the treatments of Fig. 6 a), c), respectively. It can be seen from Figure 6 a), c) that the original RSS collected by different devices at the same AP is significantly different. Especially in Fig. 6 a), the average RSS difference of all APs of Lenovo pad and Dell laptop is about 19.4dBm, the smallest difference is 2dBm at AP4, and the biggest difference is 22dBm at AP6. Although the curves of the Huawei WAS-AL00 smartphone and the Dell laptop are very similar, there is still an average RSS difference of 7.5dBm between them. Similarly, there is such a difference in the RSS data collected by the different mobile devices at the same AP in Fig. 6 c).

In addition, the fluctuation trend of the RSS curves collected by different devices is consistent. This means that RSS samples collected by different devices have some basic information in common at the same reference position. This is why the RSS collected by the same mobile device from nearby APs is higher than the RSS value collected from distant APs. The purpose of normalizing and discretizing the data in the temporary database is to map all RSS values to a uniform range while preserving the necessary information of the RSS sample, to avoid RSS differences affecting the localization system. In Fig. 6 b), d), the curves of the processed RSS data are very similar, and the values on the corresponding curves are the same for some APs. The impact of different devices on the RSS sample is significantly reduced when the processed data and the original data curve maintain the same trend, which indicates that the basic information of the original RSS sample is retained.

On the other hand, the variation of RSS values collected by heterogeneous devices at different reference positions at the same AP is shown in Fig. 7. The curve of original RSS value at AP4 and AP10



a) The original RSS value of Ap around the reference position 2 b) Processing results of Figure 6 a)



c) The original RSS value of Ap around the reference position 4 d) Processing result of Figure 6 c)

Fig. 6. RSS values collected by different devices at the same AP. a) The original RSS value of Ap around the reference position 2 b) Processing results of Figure 6 a). c) The original RSS value of Ap around the reference position 4 d) Processing result of Figure 6 c).

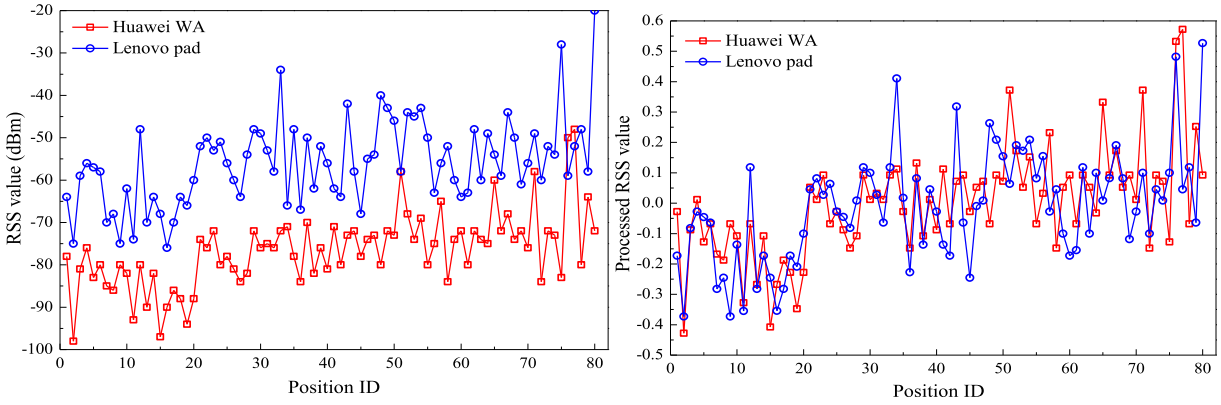
are depicted in Fig. 7 a), c), respectively. Fig. 7 b) and 7 d) are the results of the treatment of Fig. 7 a) and 7 c), respectively. In Fig. 7 a), c), at a particular AP, the original RSS collected by different devices is significantly different at each reference position. However, in Fig. 7 b), d), the curves of processed RSS data are basically consistent, which indicates that the preprocessing method can effectively avoid the difference of signal strength received by different devices.

4.1.2. The analysis of min-max processing method

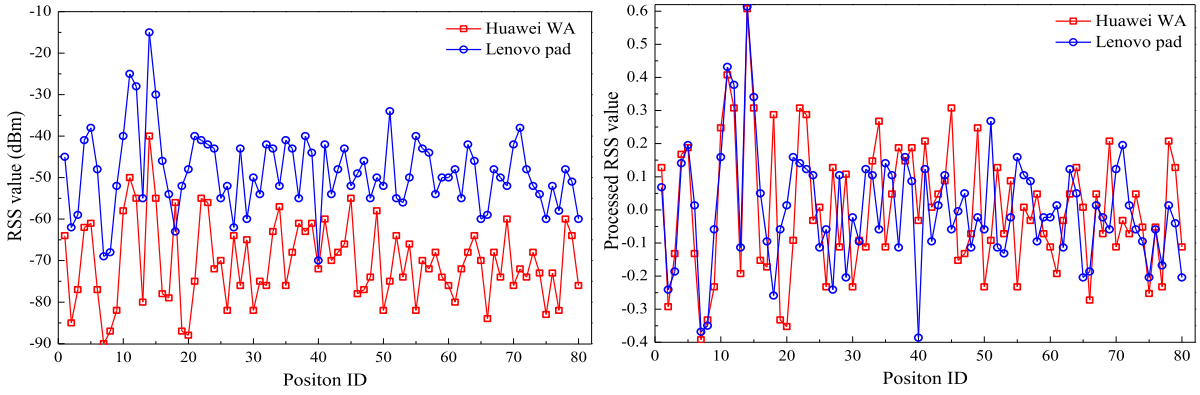
To verify the effectiveness of the min-max method, a comparison experiment between the min-max method and the existing ordering rank (OR) [34] and signal strength difference (SSD) [35] methods were performed. Eighty reference positions were randomly selected for testing on the 7th floor of the PELB. For each reference position, 60 samples with labeled position were collected as training samples by Dell laptops, Huawei smartphones and Lenovo pads, and 20 samples were collected by Samsung smartphones as test samples. Then, the min-max, the OR and SSD pre-processing methods were applied to process the training sample set and the test sample set. The position estimation algorithm obtained by training the training sample set was used to predict the test sample.

In addition, an original RSS sample set that has not been processed is added for error comparison. The *k*-NN, NB, and DT algorithms are chosen for position estimation. The effects of each method are evaluated by comparing the localization accuracy of various methods applied to different position estimation algorithms. The Cumulative Distribution Function (CDF) of the error distance is chosen to measure the localization error. The function is calculated based on the Euclidean distance between the result of predicted coordinate and the actual coordinate. The comparison results of CDF of *k*-NN, NB and DT are shown in Fig. 8 a), b) and c), respectively.

It can be seen from Fig. 8 that min-max is superior to other methods regardless of the estimation algorithm. The probability that min-max has an error distance of less than 5 m is 12%, 21%, and 13% higher than RSS, respectively. Both the OR and the SSD preprocessing methods improve overall accuracy compared to the RSS method, which illustrates the importance of using preprocessing methods. Although the curves of min-max and OR are similar in Fig. 8 a) and b), OR takes up more storage space. The SSD method is inferior to min-max and OR, because the SSD method can only preserve the difference between adjacent APs in order to reduce the storage, which will lead to the loss of some useful information, affecting the effect. In Fig. 8 c), the min-max method is significantly better than the other methods. Good performance is demonstrated in the sample set that handles the decision tree model. In addition, the final value of *k* in the *k*-NN in the above experiment is 4.



a) The original RSS changes with the observed position at AP4 b) Processing result of Figure 7 a)



c) The original RSS changes with the observed position at AP10 d) Processing result of Figure 7 c)

Fig. 7. RSS value changes with observation position. a) The original RSS changes with the observed position at AP4 b) Processing result of Figure 7 a). c) The original RSS changes with the observed position at AP10 d) Processing result of Figure 7 c).

4.2. The analysis of indoor localization accuracy

4.2.1. The analysis of localization accuracy of reconstruction algorithm

The simulation scenario of the reconstruction algorithm is shown in Fig. 9. There are 3 APs in the indoor area with a localization area of 21m × 15m, and the reference position interval is 1.5m as shown in Fig. 9 a). Gaussian noise with a mean of 0 and a variance of 1 is set in the simulation environment as shown in Fig. 9 (b). The initial fingerprint database data are generated, and then the noise variance is set to 4 under the same conditions to simulate the fingerprint database data after the environment change. To reconstruct the RSS data, eighty reference positions are selected to generate a new sample point as a new fingerprint to update the fingerprint database. These 80 new fingerprints cover the entire localization area as much as possible to ensure better reconstruction performance. The performance of the reconstruction algorithm is evaluated by calculating the difference between the reconstructed value $\widehat{rss_n}$ and the actual value rss_n^r at the n-th reference position, as shown in Eq. (27).

$$\varepsilon_n = |\widehat{rss_n} - rss_n^r| \tag{27}$$

where ε_n represents the error.

In addition, the basic localization algorithm k-NN is selected to perform the locating in the original fingerprint database, the reconstructed fingerprint database, and the real fingerprint database, in order to verify the reconstruction performance. The reconstruction algorithm is simulated 50 times, and the reconstructed signal average is taken to prevent random interference.

The results of RSS reconstruction for a single AP are shown in Fig. 10, the red triangle represents the initial RSS value of 30 points extracted from the fingerprint database; the blue square is the reconstructed RSS value; and the green circle is the real RSS value. It is observed that the reconstructed RSS value is very close to the actual changing value. The reconstruction error mean is about 1.5310dBm, and most of the reconstructed RSS values are closer to the real RSS values. This shows that in the simulation environment of this paper, the link between different reference positions can be effectively measured, because the perceptual matrix based on similarity combines the physical distance information between reference positions and the received signal strength difference information, and considers the volatility of RSS measured at a certain reference position. At the same time, the combination of Bayesian compressed sensing theory can ensure the accuracy of offline fingerprint database reconstruction.

The cumulative probability distribution of the localization error of the original fingerprint database and the fingerprint database updated using the reconstruction scheme are shown in Fig. 11. The curve of the cumulative error distribution of the localization error using the reconstruction algorithm is above the curve of the fingerprint database without updating, and is close to the curve of the real fingerprint database. Within 2m, the proportion of the localization error using the reconstruction scheme is 94%, the proportion of the localization error using the real fingerprint

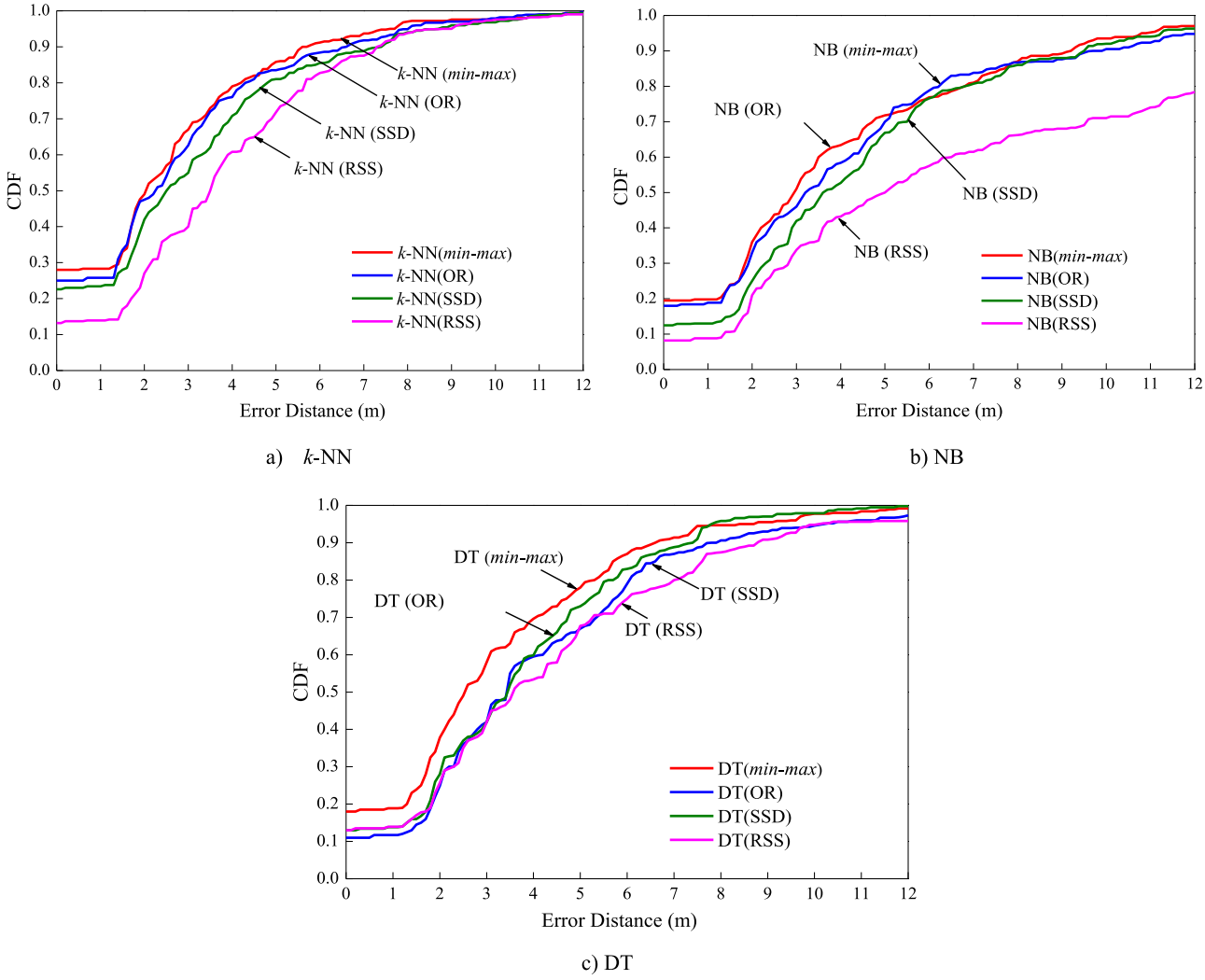


Fig. 8. CDF of the error distance of different processing algorithms.

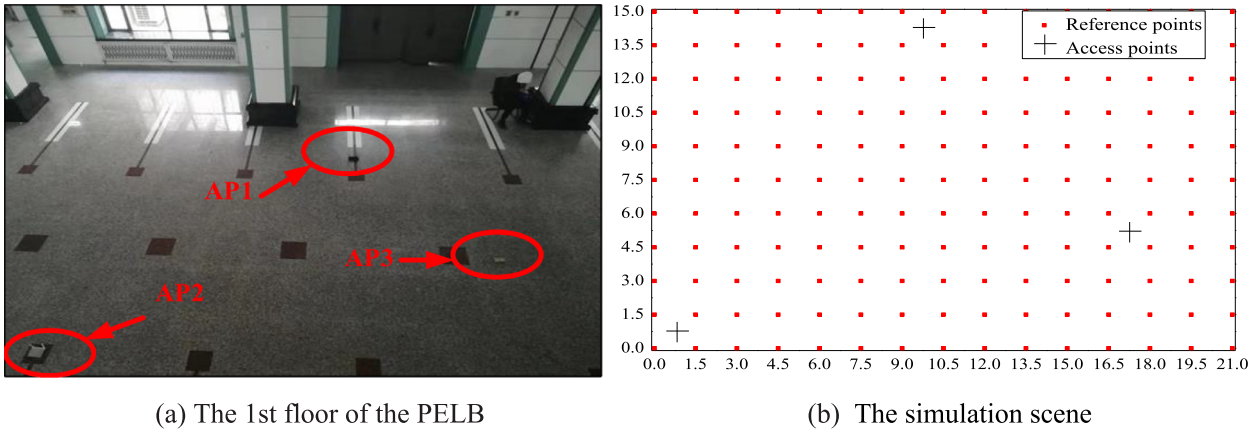


Fig. 9. The simulation scene of 21m×15m. (a) The 1st floor of the PELB. (b) The simulation scene.

database is 96%, and the proportion of the localization error using the fingerprint database without updating is only 71%. The fingerprint database constructed by the reconstruction algorithm contains some errors, leading the accuracy is lower than the real fingerprint database, but much higher than the fingerprint database without updating. Therefore, the Bayesian compressed sensing reconstruction algorithm based on similarity proposed in this paper effectively reconstructs the offline fingerprint database by collecting a small number of new fingerprints, which reduces the workload of constructing fingerprint database. The working efficiency of constructing database is improved while the impact on localization performance due to environmental changes is reduced.

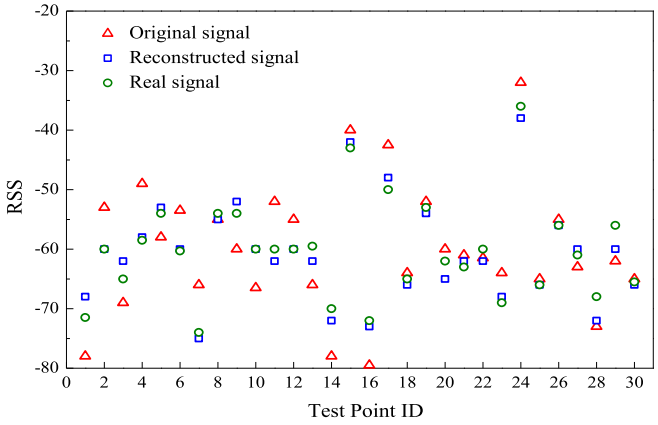


Fig. 10. Comparison of the original database and the reconstructed database.

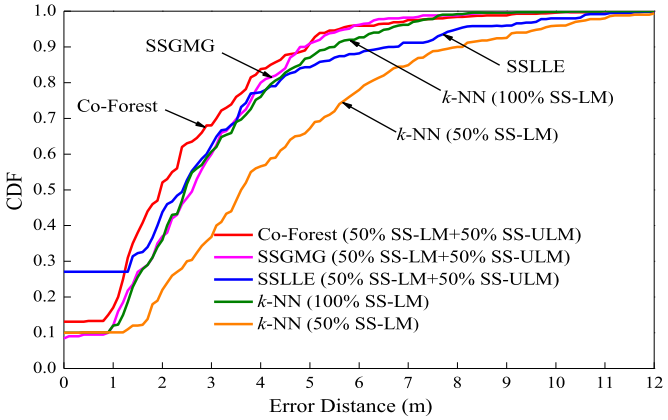


Fig. 12. CDF of the error distance of the four algorithms.

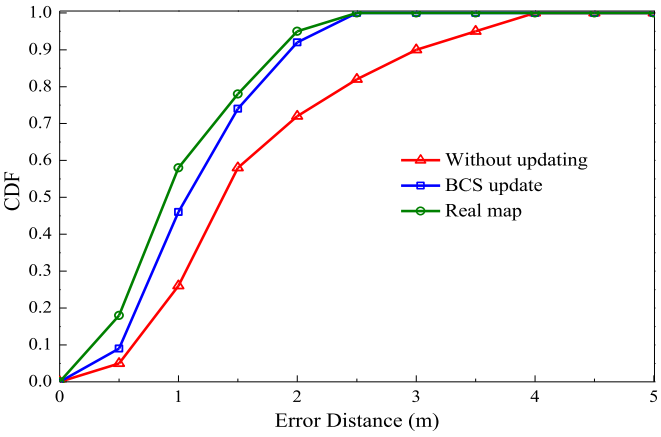


Fig. 11. CDF of different fingerprint databases.

4.2.2. The analysis under normal circumstances

In order to evaluate the performance of the proposed algorithm ILM-CFBCS, the algorithm is compared with *k*-NN and two semi-supervised learning algorithms SSLLE and SSGMG. *k*-NN is a supervised learning algorithm that uses only samples with labeled position. According to the research in this paper, the *k*-NN can achieve the best accuracy when the *k* value is 4 in this data set. The nearest neighbor *k* is set to 5 like Jain et al in implementing the SSLLE method. And the number *N* of trees in the Co-Forest in the ILM-CFBCS algorithm is 30.

The three semi-supervised algorithms SSLLE, SSGMG, and Co-Forest use both samples with labeled position and samples without labeled position. First, the samples with labeled position and the samples without labeled position are selected from the experimental sample set for training. There is a total of 6400 samples and 80 reference positions in the sample set with labeled position (80 samples with labeled position collected at each reference position). There is a total of 3200 samples without labeled position collected by volunteers. Then, all the samples participating in the training are normalized and discretized by the min-max preprocessing algorithm. Finally, 600 test samples are collected (60 reference positions are selected as test positions and 10 samples are collected at each reference position).

For ease of comparison and analysis, the three semi-supervised algorithms use only half of the sample set with labeled position and 3200 samples without labeled position. The *k*-NN algorithm runs on both the semi-labeled sample set and the full-labeled sample set. The results of CDF for the error distance of these algorithms are depicted in Fig. 12. And the correlation errors of these algorithms are compared in Table 6.

In Fig. 12, SS-LM represents the sample sets with labeled position, and SS-ULM represents the sample sets without labeled position. It can be seen from the figure that the localization performance of several semi-supervised learning algorithms using only half of the samples with labeled position and the samples without labeled position can reach or exceed the *k*-NN algorithm applying the full-labeled sample set. The localization performance of Co-Forest is superior to SSLLE and SSGMG algorithms. The error distance probability of Co-Forest within 3m is 9% and 8% lower than SSGMG and SSLLE, respectively. And the mean error is reduced by 11% and 19%, respectively. In addition, the maximum error distance of SSLLE is higher than that of Co-Forest and SSGMG, but the error distance probability is significantly reduced after 5m.

4.2.3. Analysis under special circumstances

On the one hand, the sampling result of the RSS sample may be unstable or even incomplete when the Wi-Fi signal is masked, resulting in a reduction in the performance of the algorithm. In response to this special case, the performance evaluation of the proposed algorithm was carried out in an indoor environment with multiple people moving. Several staff members conducted a one-week test sample collection in the corridor to estimate the positions and compare the mean error distances of these algorithms. A total of 500 test samples were collected, covering almost all corridors on the 7th floor of the PELB. The localization performance of the algorithms in an environment of multi-person moving is described in Table 7. The localization performance of the four algorithms is reduced compared to the "mean error distance" in Table 6. The mean error distances of Co-Forest and SSGMG increased by 50% and 64.2%, respectively. The performance degradation of *k*-NN and SSLLE is more obvious, and the mean error distance increased by 86% and 91.1%, respectively. However, the error distance of Co-Forest can be within 4m, which is better than the other three machine learning algorithms.

On another hand, the changes in the wireless LAN environment, such as AP addition or removal, can have an impact on localization performance. For this particular case, a different number of APs are used in the training sample set to compare the performance of the algorithms to estimate their robustness. The training sample used in the experiment has 25 APs with sufficient localization information. The number of APs in the training sample set is set to 5, 10, 15, 20, and 25 for experimentation, and the mean error distance of algorithms is recorded separately. The relationship between the mean error distance and the number of APs in the training sample set of the four machine learning algorithms is depicted in Fig. 13. It can be seen that the mean error distance of

Table 6
Comparison of various errors of the four algorithms.

Algorithm	Mean error (m)	Maximum error (m)	Minimum error (m)	Median error (m)
k-NN (50% SS-LM)	4.04	11.68	0	3.58
k-NN (100% SS-LM)	2.86	8.13	0	2.44
SSLLE	2.93	11.76	0	2.41
SSGMG	2.74	8.51	0.35	2.62
Co-Forest	2.44	8.02	0	1.94

Table 7
Mean error distance in the case of multi-person movement.

Algorithm	Mean error (m)
k-NN (100% SS-LM)	5.32
SSLLE	5.60
SSGMG	4.50
Co-Forest	3.66

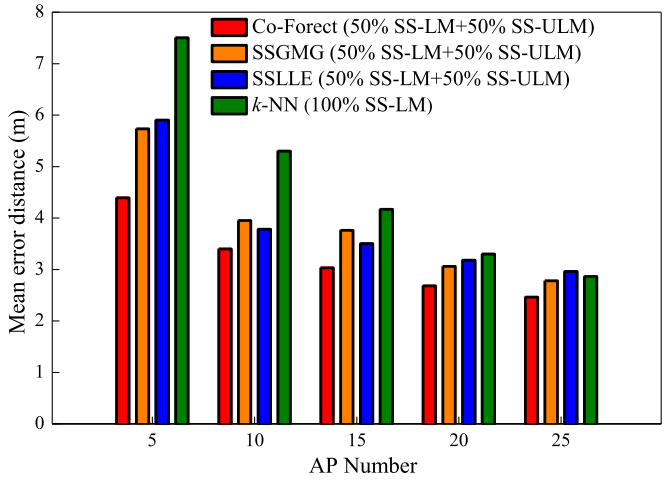


Fig. 13. The relationship between the mean error distance of the algorithm in the training sample set and the number of APs.

the four algorithms increases as the number of APs decreases. The localization performance of the four algorithms is significantly reduced when the number of APs drops from 25 to 5. The mean error distance of *k*-NN increases from 2.82m to 7.46m, and the mean variation distance is 1.16m. The mean error distance of Co-Forest increases from 2.42m to 4.44m, and the mean variation distance is 0.505m. The mean variation distances of SSGMG and SSLLE are 0.75m and 0.778m, respectively. Therefore, the localization performance of Co-Forest is least affected compared with the other three algorithms when the number of APs changes, which reflects the robustness of the proposed algorithm.

5. Conclusion

Most Wi-Fi-based indoor localization methods are time-consuming and labor-intensive in data collection and database update, and there are data differences caused by different mobile devices in data collection, which brings great limitations to the practical application. An indoor localization method based on semi-supervised algorithm Co-Forest and Bayesian Compressed Sensing (ILM-CFBCS) is proposed, which consists of offline processing stage and online localization stage. In the offline phase, the crowdsourcing technology is applied to collect data by the mobile terminal, the min-max preprocessing method is adopted to normalize and discretize the collected data, and the fingerprint database is constructed by the processed data. In the online phase, according to the RSS value sent by the user, the decision results of the random forest classifiers trained by the Co-Forest algorithm and the majority principle are used to obtain the localization result, completing the real-time location for the user.

In addition, a method for constructing offline fingerprint database is proposed, which combines the Bayesian compressed sensing theory and the similarity between reference point fingerprint. The similarity between reference points is made full use, reconstructing the changing signal from an AP in the entire localization area due to environmental changes by measuring the change of signal strength of a small number of reference points in a new environment. The fingerprint data of the new environment can be obtained by correcting the fingerprint of the original fingerprint database with the changing value.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] X. Liu, M. Jia, X.Y. Zhang, A Novel Multichannel Internet of Things Based on Dynamic Spectrum Sharing in 5G Communication, IEEE IoT J. 6 (4) (2019) 5962–5970.
[2] D. Milioris, G. Tzagkarakis, A. Papakonstantinou, et al., Low-dimensional signal-strength fingerprint-based positioning in wireless LANs, Ad Hoc Netw. 12 (2014) 100–114.
[3] O. Woodman, R. Harle, Pedestrian localization for indoor environments, in: [Proceedings of the Proceedings of the 10th international conference on Ubiquitous computing, ACM, 2008.
[4] S. Saloni, A. Hegde, Wi-Fi-aware as a connectivity solution for IoT pairing IoT with Wi-Fi aware technology: Enabling new proximity-based services, International Conference on Internet of Things & Applications, IEEE, 2016.
[5] Z. Tian, X. Tang, Z. Mu, et al., Fingerprint indoor positioning algorithm based on affinity propagation clustering, Eurasip J. Wireless Commun. Netw. 2013 (1) (2013) 272–281.
[6] X. Liu, X.Y. Zhang, "Rate and Energy Efficiency Improvements for 5G-Based IoT With Simultaneous Transfer, IEEE IoT J. 6 (4) (2019) 5971–5980.
[7] S. Kumar, R.M. Hegde, N. Trigoni, Gaussian process regression for fingerprinting based localization, Ad Hoc Netw. 51 (2016) 1–10.
[8] M. Ochiai, M. Fujii, A. Ito, et al., A study on indoor position estimation based on fingerprinting using GPS signals, 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN), IEEE, 2014.
[9] X. Li., A GPS-based Indoor Positioning System with Delayed Repeaters, IEEE Trans. Vehic. Technol. (2018) 1.
[10] Y. Gu, C. Zhou, A. Wieser, et al., Trajectory estimation and crowdsourced radio map establishment from foot-mounted IMUs, Wi-fi fingerprints and GPS positions, IEEE Sens. J. (2018) 1.
[11] K.Y. Qiu, H. Huang, W. Li., Indoor geomagnetic positioning based on a joint algorithm of particle filter and dynamic time warp, 2018 Ubiquitous Positioning, Indoor Navigation and Position-Based Services (UPINPBS), IEEE, 2018.
[12] S.C. Yeh, W.H. Hsu, W.Y. Lin, Study on an indoor positioning system using earth's magnetic field, IEEE Transactions on Instrumentation and Measurement, IEEE, 2019.
[13] W.B. Shen, Research on multi-information fusion indoor positioning technology based on geomagnetic fingerprint and inertial sensor, Guangzhou: South China Univ. Technol. (2017).
[14] C. Whitelamand, T. Bourlai, Accurate eye localization in the short waved infrared spectrum through summation range filters, Computer Vis Image Und 139 (2015) 59–72.
[15] C Medina, J.C. Segura, A. De la Torre, Ultrasound indoor positioning system based on a low-power wireless sensor network providing sub-centimeter accuracy, Sensors 13 (3) (2013) 3501–3526.
[16] K. Pahlavan, P. Krishnamurthy, Y. Geng, Localization challenges for the emergence of the smart world, IEEE Access 3 (2015) 3058–3067.

[17] P. Deng, P.Z. Fan, An AOA assisted TOA positioning system, in: In Proceedings of the WCC-ICCT International Conference on Communication Technology Proceedings, Beijing, China, August 2000, pp. 1501–1504. 21–25.
[18] N. Patwari, J.N. Ash, S. Kyperountas, Locating the nodes: cooperative localization in wireless sensor networks, IEEE Signal Process. Mag. 22 (2005) 54–69.
[19] N.B. Priyantha, A. Chakraborty, H. Balakrishnan, The Cricket position-support system, in: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, Boston, MA, USA, August 2000, pp. 32–43. 6–11.
[20] K. Srinivasan, P. Levis, K. Srinivasan, RSSI is under appreciated, in: [C]. In Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets); Stanford Information Networks Group, Stanford, CA, USA, 2006.
[21] A. Khalajmehrabadi, N. Gatsis, D. Akopian, Modern WLAN fingerprinting indoor localization methods and deployment challenges, IEEE Commun. Surv. Tutor. 19 (3) (2017) 1974–2002.
[22] S. He, S.H.G. Chan, Wi-Fi fingerprint-based indoor localization: recent advances and comparisons, IEEE Commun. Surv. Tutor. 18 (1) (2017) 466–490.
[23] G.S. Wu, P.H. Tseng, A deep neural network-based indoor positioning method using channel state information, 2018 International Conference on Computing, Networking and Communications (ICNC), 2018.
[24] R. Ma, Q. Guo, C. Hu, et al., An improved Wi-Fi indoor localization algorithm by weighted fusion, Sensors 15 (9) (2015) 21824.
[25] R. Battiti, T.L. Nhat, A. Villani, Position-Aware Computing: A Neural Network Model for Determining Position in Wireless LANs, University of Trento, 2002 Technical report DIT-02-0083.
[26] X.L. Gan, B.G. Yu, L. Huang, Deep learning for weights training and indoor positioning using multi-sensor fingerprint, [2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), IEEE, 2017.
[27] V.K. Jain, S. Tapaswi, A. Shukla, Position estimation based on semi-supervised locally linear embedding (SSLLE) approach for indoor wireless networks, Wireless Personal Commun. 67 (4) (2012) 879–893.
[28] S.B. Li, S.Z. Wang, X. Zhang, et al., Semi-supervised learning indoor positioning algorithm based on global manifold structure, Comput. Modern. 7 (2019) 82–87.
[29] C. Feng, Research and Implementation of RSS Indoor Positioning System Based on Compressed Sensing, Beijing Jiaotong University, 2010.
[30] http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm.
[31] C. Feng, W.S.A. Au, S Valaee, et al., Received-signal-strength-based indoor localization using compressive sensing, IEEE Trans. Mobile Comput. 11 (12) (2012) 1983–1993.
[32] S. He, S.H.G. Chan, L. Yu, et al., Fusing noisy fingerprints with distance bounds for indoor localization, Computer Commun. IEEE (2015) 2506–2514.
[33] M. E. Tipping. "Sparse bayesian learning and the relevance vector machine" JMLR.org, 2001.
[34] K. Wooseong, Y. Sungwon, G. Mario, et al., Crowdsource based indoor localization by uncalibrated heterogeneous Wi-Fi devices, Mobile Inf. Syst. 2016 (2016) 1–18.
[35] A.K.M. Mahtab Hossain, Y. Jin, W.S. Soh, et al., SSD: A robust RF position fingerprint addressing mobile devices heterogeneity, IEEE Trans. Mobile Comput. 12 (1) (2013) 65–77.

Min Zhao, born in 1995. She received her B. Sc. degree in communication engineering from Heilongjiang University in 2018. Currently, she studys at the department of Communication Engineering of Heilongjiang University, Harbin, P.R.China. Her researches include wireless sensor network, wireless multi-hop routing and ubiquitous sensing.

Danyang Qin is a postdoctoral fellow in Electronic Science and technology Post-Doctoral Research Center and an associated professor in Heilongjiang University. She received her B. Sc. degree in communication engineering from Harbin Institute of Technology in 2006, and both M. Sc and Ph. D. degree in information and communication system from Harbin Institute of Technology in 2008 and 2011 respectively. Her current researches include wireless sensor network, wireless multi-hop routing and ubiquitous sensing.

Ruolin Guo, born in 1996. She received her B. Sc. degree in electronic information engineering from Heilongjiang University in 2018. Currently, she studys at the department of Communication Engineering of Heilongjiang University, Harbin, P.R.China. Her researches include wireless sensor network, wireless multi-hop routing and ubiquitous sensing.

Guangchao Xu, born in 1994. He received his B. Sc. degree in electronic information engineering from Changchun Industrial University in 2018. Currently, he studys at the department of Communication Engineering of Heilongjiang University, Harbin, P.R.China. His researches include wireless sensor network, wireless multi-hop routing and ubiquitous sensing.